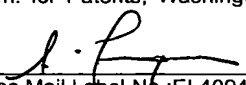


I hereby certify that this paper is being deposited with the United States Postal Service as Express Mail in an envelope addressed to: Asst. Comm. for Patents, Washington, D.C. 20231, on this date.

01/18/00
Date


Express Mail Label No.: EL409490885US

APPLICATION FOR
UNITED STATES LETTERS PATENT
SPECIFICATION

Inventor(s): Junichi HAGIWARA, Tatsuya SHINDO,
and Katsuhiko TAGUCHI

Title of the Invention: SEARCH SYSTEM AND METHOD BASED ON
SEARCH CONDITION COMBINATIONS

SEARCH SYSTEM AND METHOD BASED ON SEARCH CONDITION COMBINATIONS

Background of the Invention

5 Field of the Invention

The present invention relates to a search system for searching for the information of a document, etc. by combining a plurality of search conditions, and a method thereof.

10

Description of the Related Art

A full text search system is used to extract a text including a target keyword from a large volume of text data. Examples of application fields include an Internet search, a patent search, a document search by groupware, a document search in a library, etc.

Sub The full text search system holds document information to be searched as a text database in a form referred to as an inverted index. When a search expression where a plurality of keywords are combined by a logical operator such as "AND" or "OR" is provided as an input, the number of documents and a list of (the names or the IDs of) the documents, which satisfy the condition represented by the search

expression, are returned as search results.

If keywords are combined by "AND", a document including both of the combined keywords is returned. If keywords are combined by "OR", a document including at least either of the keywords is returned. Additionally, a related word which frequently appears along with a keyword included in an input search expression can be returned depending on a system.

10 Fig. 1A shows such a full text search system. The system shown in Fig. 1A comprises a text database 1 and a full text search engine 2. Assuming that a search expression "TOKYO AND SUBWAY" is input, the search engine 2 searches the text database 1 for a document including both of the keywords "TOKYO" and "SUBWAY", and outputs a search result. Here, the number of hit documents, the IDs of the hit documents, and words related to the keywords are output.

20 Fig. 1B shows a typical user interface for inputs/outputs in the full text search system shown in Fig. 1A. On a search condition input screen 3, a database to be searched can be selected from three databases DB1, DB2, and DB3, and a search expression
25 of keywords (Keywords) and duration to be searched

(DATE) can be specified. The duration to be searched indicates the range of the dates written in documents.

In this example, DB2 is selected as a database
5 to be searched, "TOKYO AND SUBWAY" is specified as a search expression, and the dates from October 21, 1995 till June 1, 1996 are specified as duration to be searched.

When a user presses (clicks with a pointing
10 device) a Search button 4 after setting these search conditions, a search process starts to be executed. On a search result output screen 5, the number of documents found as a result of the search and the list of the documents are displayed. In this example,
15 the finding of 4 documents, the list of the IDs of the documents (DocID), and their titles (Title) are displayed.

Furthermore, on a related word output screen 6,
a list of keywords related to search conditions is
20 displayed. In this example, keywords such as "Marunouchi-line", etc. are displayed as related words.

However, the above described conventional full text search system has the following problems.

25 With a conventional user interface, the number

of search condition combinations that can be issued at one time is only one. If a search result change is desired to be examined by slightly changing respective search conditions, a search must be repeated the number of times, which is equal to the number of combinations of different search conditions.

For example, if a search is desired to be performed with the search expression "TOKYO AND SUBWAY" by dividing the duration from 1992 till 1998 in year units, a search must be repeated 7 times while changing the condition of the duration to be searched. Accordingly, a heavy workload of inputting the search conditions is imposed, which leads to system operability difficulties for the user.

Furthermore, since search results output for a plurality of search condition combinations are displayed for the respective combinations, it is difficult to arrange and compare the results. As a result, a search result change when a search condition is slightly changed cannot be recognized inclusively.

Summary of the Invention

An object of the present invention is to provide

a search system for realizing an improved user interface when search results are obtained for respective search condition combinations in an information search such as a full text search, etc.

5 In a first aspect of the present invention, a search system comprises an inputting unit and an instructing unit. The inputting unit inputs the specification information for collectively specifying a plurality of search condition combinations. The
10 instructing unit instructs an information search based on the input specification information.

 In a second aspect of the present invention, a search system comprises an obtaining unit and an outputting unit. The obtaining unit obtains a
15 plurality of search results for a plurality of search condition combinations. The outputting unit collectively outputs the output information corresponding to the plurality of obtained search results.

20 In a third aspect of the present invention, a search system comprises an inputting unit, a searching unit, and an outputting unit. The inputting unit inputs the specification information for collectively specifying a plurality of search
25 condition combinations. The searching unit performs

an information search based on the specification information. The outputting unit collectively outputs the output information corresponding to a plurality of search results for the plurality of search condition combinations.

Brief Description of the Drawings

Fig. 1A is a schematic showing a conventional full text search system;

10 Fig. 1B is a schematic showing a conventional user interface;

Fig. 2A is a block diagram showing the principle of a search system according to the present invention;

15 Fig. 2B is a block diagram showing the configuration of a full text search system;

Fig. 3 shows a multi-dimensional input screen;

Fig. 4 shows an input/output screen in the form of a table;

20 Fig. 5 shows a multi-dimensional output screen;

Fig. 6 shows a related word display;

Fig. 7 shows a search result list display;

Fig. 8 shows a search condition change;

Fig. 9 shows a first line graph;

25 Fig. 10 shows a graph display and a list

display;

Fig. 11 shows a second line graph;

Fig. 12 shows a first bar chart;

Fig. 13 shows a second bar chart;

5 Fig. 14 shows a third bar chart;

Fig. 15 shows pie charts;

Fig. 16 is a flowchart showing the process performed by a multi-dimensional input module;

Fig. 17 shows a first input array;

10 Fig. 18 shows a second input array;

Fig. 19 is a flowchart showing the process performed by an automatic search expression generation module;

Fig. 20 shows a search array;

15 Fig. 21 is a flowchart showing the process performed by a search result switching module;

Fig. 22 is a flowchart showing the process performed by a multi-dimensional output module;

20 Fig. 23 is a flowchart showing the process performed by a visualizing module;

Fig. 24 is a block diagram showing the configuration of an information processing device; and

Fig. 25 shows storage media.

Description of the Preferred Embodiments

Details of preferred embodiments according to the present invention are explained below by referring to the drawings.

5 Fig. 2A is a block diagram showing the principle of a search system according to the present invention.

10 In the first aspect of the present invention, the search system comprises an inputting unit 11 and an instructing unit 12. The inputting unit 11 inputs the specification information for collectively specifying a plurality of search condition combinations. The instructing unit 12 instructs an information search based on the input specification information.

15

20 The instructing unit 12 instructs the information search specified by the specification information based on the specification information input by the inputting unit 11. As a result, the information search is automatically performed for a plurality of search condition combinations.

25 With such a search system, when a search is performed for various combinations of a plurality of search conditions such as a database to be searched, duration to be searched, a keyword, etc., a user can

collectively input these combinations. Accordingly, an input interface easy to use for the user can be realized, which leads to an increase in an information search efficiency.

5 In the second aspect of the present invention, the search system comprises an obtaining unit 13 and an outputting unit 14. The obtaining unit 13 obtains a plurality of search results for a plurality of search condition combinations. The outputting unit 14
10 collectively outputs the output information corresponding to the plurality of obtained search results.

 The outputting unit 14 generates output information from a plurality of search results
15 obtained by the obtaining unit 13, and outputs the obtained output information. As a result, the information of a plurality of search results obtained by the search performed for a plurality of search condition combinations are automatically output.

20 With such a search system, a plurality of obtained search results are collectively output when a search is performed for various combinations of a plurality of search conditions, whereby a user can easily compare the plurality of search results.
25 Accordingly, differences between search results can

be inclusively recognized, so that an output interface easy to use for a user can be realized.

As described above, one point of the present invention is to collectively specify a plurality of search condition combinations, and another is to collectively output a plurality of search results for the plurality of search condition combinations.

For example, the inputting unit 11 shown in Fig. 2A corresponds to a multi-dimensional input module 25 which is shown in Fig. 2B and will be described later, the instructing unit 12 shown in Fig. 2A corresponds to an automatic search expression generation module shown in Fig. 26 shown in Fig. 2B, the obtaining unit 13 shown in Fig. 2A corresponds to a search result switching module 27 shown in Fig. 2B, and the outputting unit 14 shown in Fig. 2A corresponds to a multi-dimensional output module 28 and a visualizing module 29, which are shown in Fig. 2B.

Fig. 2B is a block diagram showing the configuration of a full text search system according to a preferred embodiment. The system shown in Fig. 2B comprises a search device 21 and an interface 22. The search device 21 corresponds to, for example, a conventional full text search system. This device

comprises a text database 23 and a full text search engine 24.

The interface 22 comprises the multi-dimensional input module 25, the automatic search expression generation module 26, the search result switching module 27, the multi-dimensional output module 28, and the visualizing module 29.

The multi-dimensional input module 25 receives the information for specifying a plurality of search condition combinations from a user in a multi-dimensional data format, and inputs the received information to the automatic search expression generation module 26. The automatic search expression generation module 26 generates from the given information a plurality of search condition combinations (search expressions) that can be accepted by the search device 21, and sequentially provides the search device 21 with the generated combinations one by one. Upon receipt of each of the search expressions, the search engine 24 searches the text database 23 for each of them, and outputs a search result to the interface 22 for each.

The search result switching module 27 receives a plurality of search results from the search device 21, and selects necessary data as output information.

The multi-dimensional output module 28 outputs the data selected by the search result switching module 27 in a multi-dimensional data format.

5 The visualizing module 29 visualizes and outputs a plurality of search results in the form of a graph. Additionally, the visualizing module 29 can also extract a particular search result specified by a user on the graph from the search result switching module 27, and can selectively output more detailed
10 information. If the number of hit items of information is large and detailed information cannot be kept in the search result switching module 27, the particular search expression corresponding to the specified search result is again input to the search
15 device 21. Then, the visualizing unit 29 receives a detailed search result from the search engine 24, and outputs the result.

The multi-dimensional input module 25 comprises a changed portion extracting unit 30. Accordingly,
20 the multi-dimensional module 25 can also extract a changed portion of input multi-dimensional data, and can input the information of the extracted portion to the automatic search expression generation module 26. In this case, the automatic search expression
25 generation module 26 automatically generates only a

search expression including the changed condition, and provides the search device 21 with the generated search expression. In the meantime, the multi-dimensional output module 28 comprises a changed
5 portion reflecting unit 31. This module reflects a search result based on a changed condition on already output information, and modifies the output information.

If the search device 21 corresponds to an
10 independent full text search system, the configuration shown in Fig. 2B can be implemented by providing the interface 22 as a system externally attached to the search device 21. Furthermore, the search device 21 and the interface 22 are integrated
15 into one system to be provided.

In a client-server system, the search device 21 and the interface 22 may be respectively arranged in a server and a client. In this case, the search device 21 and the interface 22 are interconnected via
20 a communications network.

Next, specific examples of the processes performed by the system shown in Fig. 2B are explained by referring to Figs. 3 through 15.

Fig. 3 exemplifies a multi-dimensional input
25 screen that the multi-dimensional input module 25

generates on a display. Here, a dimension is set for each search condition type, and a plurality of elements can be set for each dimension. The plurality of elements are vertically arranged for each dimension, and "AND" is used to make a correspondence between dimensions. A user can specify necessary data from a menu, or may directly input necessary data.

First of all, when a user specify the number of dimensions used for a search in an input box 41, input boxes 42 and 43 corresponding to the specified number of dimensions are displayed. Here, "4" is specified as the number of dimensions, and respective 4 input boxes 42 and 43 are displayed in correspondence with the first through the fourth dimensions ("1st dimension", "2nd dimension", "3rd dimension", and "4th dimension").

Next, the condition type of each of the dimensions is defined in the input box 42, and the number of elements of the condition of each of the dimensions is specified in the input box 43, so that input boxes the number of which corresponds to the specified number of elements are displayed.

Here, "Year", "Database", "Keyword", and "Keyword" are specified as the condition types of the first through the fourth dimensions. "Year" indicates

duration to be searched, "Database" indicates a database to be searched, and "Keyword" indicates a keyword included in a document.

5 Additionally, "7", "2", "2", and "2" are respectively specified as the numbers of elements in these dimensions. Accordingly, 7 input boxes 44 are displayed for the first dimension, and 2 input boxes 44 are respectively displayed for the second through the fourth dimensions.

10 At this time point, a user inputs the conditions corresponding to the respective dimensions to the input boxes 44. Here, seven conditions "1992" through "1998" are input to the input boxes 44 for the first dimension, two conditions "Newspaper-DB" and "Patent-DB" are input to the input boxes 44 for the second
15 dimension, two conditions "Fujitsu" and "I-Company" are input to the input boxes 44 for the third dimension, and two conditions "Parallel" and "WWW" are input to the input boxes 44 for the fourth
20 dimension.

When a user clicks a search button 45, a search is performed under the input search conditions for the respective dimensions. With such an input screen, it is possible to collectively instruct the issuance
25 of search expressions of all combinations which are

obtained by extracting the elements from the respective dimensions one by one. Accordingly, a user workload on the operations for inputting a plurality of search expressions can be reduced.

5 Fig. 4 exemplifies an input screen of multi-dimensional data integrated with an output screen of search results. Here, a region ("Input part") 51 enclosed by a thick line is arranged in the upper left of the input/output screen in the form of a
10 table, and search conditions are input to this region. In the remaining region ("Output part") 52, search results are displayed.

 The condition types (such as "Database", "Keyword", "Year", etc.) assigned to the respective
15 dimensions are set in the leftmost column, and one or more conditions are written to the columns at the right of the leftmost column. For the sake of representing multi-dimensional data on a two-dimensional plane, the same conditions are repeatedly
20 displayed inside the table as shown in the upper two rows of "Keyword".

 For example, "Fujitsu" and "I-Company" in the row of the first "Keyword" respectively appear twice in order to be respectively combined with "Patent-DB"
25 and "Newspaper-DB". As a result, four combinations

"Patent-DB" and "Fujitsu", "Patent-DB" and "I-Company", "Newspaper-DB" and "Fujitsu", and "Newspaper-DB" and "I-Company" are set.

5 Additionally, "Parallel" and "WWW" in the row of the next "Keyword" respectively appear four times in order to be further combined with the above described four combinations external to these keywords. Note that, however, the conditions which are actually input by a user are only within the portions enclosed
10 by thick lines, and the other repetition portions are automatically copied and displayed.

By user clicking the search button 53, a search is executed under the input search conditions of the respective dimensions. With such an input screen in
15 the form of the table, a user can visually recognize the total number of search expressions specified with multi-dimensional data according to the size of the table, and can suitably increase/decrease the number of conditions based on the size of the table.
20 Accordingly, the operations for inputting a plurality of search expressions can be made efficient.

When the multi-dimensional data is input by the multi-dimensional input module 25 in this way, the automatic search expression generation module 26
25 automatically generates search expressions the number

of which is equivalent to the total number of round robin combinations of the elements in all the dimensions. Here, assuming that the number of dimensions is "n" and the number of elements of an "i"th dimension is "e(i)", the total number of generated search expressions is given by the following equation.

$$\text{NUMBER OF SEARCH EXPRESSIONS} = \prod_{i=1}^n e(i)$$

By way of example, for the multi-dimensional data shown in Figs. 3 and 4, the number of search expressions=7*2*2*2=56. Search expressions with the "Year" condition being "1992" are the following eight expressions.

1. (Year==1992) and (Database==Newspaper-DB) and (Keyword==Fujitsu) and (Keyword==Parallel)
2. (Year==1992) and (Database==Newspaper-DB) and (Keyword==Fujitsu) and (Keyword==WWW)
3. (Year==1992) and (Database==Newspaper-DB) and (Keyword==I-Company) and (Keyword==Parallel)
4. (Year==1992) and (Database==Newspaper-DB) and (Keyword==I-Company) and (Keyword==WWW)
5. (Year==1992) and (Database==Patent-DB) and

(Keyword==Fujitsu) and (Keyword==Parallel)

6. (Year==1992) and (Database==Patent-DB) and
(Keyword==Fujitsu) and (Keyword==WWW)

7. (Year==1992) and (database==Patent-DB) and
5 (Keyword==I-Company and (Keyword==Parallel)

8. (Year==1992) and (Database==Patent-DB) and
(Keyword==I-Company) and (Keyword==WWW)

Also the search expressions with the "Year"
condition being "1993" through "1998" are similar. As
10 described above, because the automatic search
expression generation module 26 automatically
generates a plurality of search expressions from
multi-dimensional data, a user does not need to input
the search expressions one by one, which leads to a
15 reduction in a user workload.

Generated search expressions are sequentially
provided to the search device 21, and the
corresponding search results are sequentially
returned to the interface 22. These search results
20 normally include the information such as document
IDs, etc., which are not required by the multi-
dimensional output module 28. Accordingly, the search
result switching module 27 extracts from the received
search results the information of the number of hits,
25 a related word, a search result list, etc., which are

required by the multi-dimensional output module 28, and passes the extracted information to the multi-dimensional output module 28.

5 Whichever input method shown in Fig. 3 or 4 is used, the multi-dimensional output module 28 collectively outputs search results in the form of the table shown in Fig. 4, and displays the output information corresponding to each search expression in each cell in the region 52. Here, 56 cells are
10 displayed in correspondence with 56 search expressions.

At this time, a user can switch between the displays of the number of hits ("Total Hit Number"), a related word ("Related Words"), and a search result
15 list ("Hit Documents List") by clicking a selection button 54, 55, or 56. Accordingly, contents of the type selected by the user are displayed in the respective cells.

For example, if the selection button 54 is
20 clicked and activated, the numbers of hits are displayed in the respective cells as shown in Fig. 5. When the selection button 55 is activated, related words are displayed in the respective cells as shown in Fig. 6. When the selection button 56 is activated,
25 search result lists are displayed in the respective

cells as shown in Fig. 7.

By collectively displaying search results based on a plurality of search expressions as described above, search result changes when search conditions are slightly changed can be inclusively recognized and the entire tendency of the search results can be grasped. Additionally, since the type of output information can be switched, the diversified tendency of search results can be grasped.

In the meantime, when a user changes a portion of a search condition of multi-dimensional data on the input screen shown in Fig. 3 or 4 or on the output screen shown in Fig. 5, 6, or 7, and again instructs a search, the interface 22 performs a search only for the changed portion and updates output information. On the output screen shown in Fig. 5, 6, or 7, a user changes a search condition and clicks the search button 53, so that a search is again performed.

First of all, the changed portion extracting unit 30 in the multi-dimensional input module 25 makes a comparison between the condition in each dimension in the previous search and that in the current search, and puts an input mark on a changed condition. The automatic search expression generation

module 26 generates only the search expression corresponding to the difference between the multi-dimensional data in the previous and the current searches based on this mark.

5 For example, "1992" in the first dimension shown in Fig. 3 is changed to "1991", the following eight search expressions are generated.

1. (Year==1991) and (Database==Newspaper-DB) and
(Keyword==Fujitsu) and (keyword==Parallel)
- 10 2. (Year=-1991) and (Database==Newspaper-DB) and
(Keyword==Fujitsu) and (Keyword==WWW)
3. (Year==1991) and (Database==Newspaper-DB) and
(Keyword==I-Company) and (Keyword==Parallel)
4. (Year==1991) and (Database==Newspaper-DB) and
15 (Kkeyword==I-Company) and (Keyword==WWW)
5. (Year==1991) and (Database==Patent-DB) and
(Keyword==Fujitsu) and (Keyword==Parallel)
6. (Year==1991) and (Database==Patent-DB) and
(Keyword==Fujitsu) and (Keyword==WWW)
- 20 7. (Year==1991) and (database==Patent-DB) and
(Keyword==I-Company) and (Keyword==Parallel)
8. (Year==1991) and (Database==Patent-DB) and
(Keyword==I-Company) and (Keyword==WWW)

25 These search expressions are sequentially provided to the search device 21, and the

corresponding search results are sequentially returned to the interface 22. The changed portion reflecting unit 31 in the multi-dimensional output module 28 updates the portion to which a change is made as output information as a result of the search corresponding to the difference between multi-dimensional data. At this time, the multi-dimensional output module 28 again outputs only the search result corresponding to the difference by using the information representing the presence/absence of the change, which is set by the multi-dimensional input module 25.

For example, if "1992" in the first dimension is changed to "1991" as described above, only the search results in the cells in the row of "1991", which are shown in Fig. 8 and enclosed by a thick line, are rewritten.

As described above, when a portion of a search condition represented by multi-dimensional data is changed, only a search for the changed portion is selectively executed and the corresponding output information is rewritten, so that the process is significantly made efficient compared with the case where a search is repeated for all of search conditions.

Additionally, if a user instructs a graph display on a search result output screen, the visualizing module 29 generates a graph which represents the number of hits from the search results
 5 of a multi-dimensional search.

Sub 2 Supposing that a user clicks a graph button 57 on the output screen shown in Fig. 5, a line graph shown in Fig. 9 is displayed. The vertical axis of this graph represents the number of hits, while the
 10 horizontal axis represents the condition specified by a user. Here, "Year" is specified as the condition of the horizontal axis, and a line 61 represents the transition of the number of hits of each year, which corresponds to the combination classification of
 15 "(Database==Patent-DB) and (Keyword==Fujitsu)".

Furthermore, lines 62, 63, and 64 respectively correspond to the classifications such as
 "(Database==Patent-DB) and (keyword==I-Company)",
 "(Database==Newspaper-DB) and (Keyword==Fujitsu)",
 20 and "(Database==Newspaper-DB) and (Keyword==I-Company)".

Here, if a user specifies a portion of the displayed graph, the visualizing module 29 selectively displays as a list the search results
 25 corresponding to the specified portion. For example,

in the line graph shown in Fig. 9, when a user clicks a point "P" (the number of hits=61), which corresponds to the year 1996 on the line 62, a search result list 65 including 61 hit documents is displayed as shown in Fig. 10. This search result list 65 includes the information such as the titles, the summaries, etc. of the hit documents.

If the search result list 65 is stored in the search result switching module 27 at this time, the visualizing module 29 extracts this list from the search result switching module 27, and outputs the list. Additionally, if the search result list 65 is not stored in the search result switching module 27, search expressions corresponding to the specified portion in the graph are selectively generated and input to the search device 21. Then, the visualizing module 29 receives the search result list 65 from the search engine 24, and outputs the list 65.

Furthermore, the visualizing module 29 can display various graphs shown in Figs. 11 through 15 other than the line graph shown in Fig. 9. Also in these graphs, a user specifies a point or a region in the graphs, so that the search results corresponding to the specified portion are displayed as a list.

The line graph shown in Fig. 11 represents the

transition of the total number of hits of each year shown in Fig. 5. A line 71 corresponds to the line 61 shown in Fig. 9, and a line 72 corresponds to the total of the numbers of hits of the lines 61 and 62 in Fig. 9. A line 73 corresponds to the total of the numbers of the lines 61, 62, and 63 in Fig. 9. A line 74 corresponds to the total of the numbers of hits of the lines 61, 62, 63, and 64 in Fig. 9.

A bar chart shown in Fig. 12 represents the transition of the number of hits of each year shown in Fig. 5. Bars 81, 82, 83, and 84 in Fig. 12 respectively correspond to the numbers of hits of the lines 61, 62, 63, and 64 in Fig. 9.

A bar chart shown in Fig. 13 represents the transition of the total number of hits of each year in Fig. 5. The bar chart shown in Fig. 13 corresponds to the line 74 in Fig. 11. Rectangles 91, 92, 93, and 94 within each bar respectively correspond to the numbers of hits in the lines 61, 62, 63, and 64 in Fig. 9.

A bar chart shown in Fig. 14 represents the transition of the number of hits in each of the classifications in Fig. 5.

Here, the transition of the number of hits of each year is shown for each of the four

classifications corresponding to the four lines in Fig. 9. Each of bars 101, 102, 103, 104, 105, 106, and 107 within the respective classifications corresponds to one point on one line in Fig. 9.

5 Pie charts shown in Fig. 15 represent the transition of the percentage of the number of hits of each year in Fig. 5. In each of the pie charts shown in Fig. 15, sectors 111, 112, 113, and 114 respectively correspond to the bars 81, 82, 83, and
10 84 in Fig. 12.

By automatically displaying the number of hits as a graph as described above, the overall tendency of search results can be visually grasped. Additionally, since the search results corresponding
15 to the portion specified by a user are automatically displayed as a list, the user can obtain locally detailed information while viewing the graph.

Next, details of the processes performed by the respective modules shown in Fig. 2B are explained by
20 referring to Figs. 16 through 22.

Fig. 16 is a flowchart showing the process performed by the multi-dimensional input module 25. The multi-dimensional input module 25 first initializes an input screen (step S1), inputs a
25 specified number of dimensions (step S2), inputs the

number of elements in each of the specified dimensions (step S3), and initializes an input array representing multi-dimensional data (step S4).

Assuming that the number of dimensions is "4",
5 the number of elements in the first dimension is "7",
and the numbers of elements in the second through the
fourth dimensions are "2" as shown in Fig. 3, an
input array shown in Fig. 17 is generated. The length
and the width of this input array are variable. The
10 width corresponds to the double of the number of
dimensions, while the length corresponds to the
number of elements in each dimension. Furthermore, if
a portion of multi-dimensional data is changed and a
search is again performed, the input array of the
15 multi-dimensional data is used unchanged.

Next, the multi-dimensional input module 25 sets
all of search conditions to the state where no
change is made by clearing an input mark on the input
array (step S5), inputs the respective search
20 conditions in the respective dimensions (step S6),
and registers the input search conditions to the
input array (step S7). At this time, the changed
portion extracting unit 30 puts input marks on the
registered search conditions, and sets them to the
25 state where a change is made.

Next, the multi-dimensional input module 25 determines whether or not the input of the search conditions is completed (step S8). If the input is not completed, the multi-dimensional input module 25
5 repeats the operations in and after step S6. For example, if a user inputs the next search condition, the input of the search conditions is determined not to be completed.

With such a process, when all of search
10 conditions in an input array are newly input, input marks are put on all of the search conditions as shown in Fig. 17. If only some of the search conditions are newly input, input marks are put only on the input search conditions as shown in Fig. 18.

15 Upon completion of the input, the multi-dimensional input module 25 then determines whether or not search execution is instructed (step S9). If the instruction is not issued, the multi-dimensional input module 25 further determines whether or not to
20 continue the input process (step S10). If the multi-dimensional input module 25 determines to continue the input process, it repeats the operations in and after step S5. If the multi-dimensional input module 25 determines not to continue the input process, it
25 terminates the process. Whether or not to continue

the input process is determined, for example, based on the instruction from the user.

If the search execution is instructed in step S9, the multi-dimensional input module 25 passes the information of the input array to the automatic search expression generation module 26, requests the automatic generation of a search expression (step S11), and terminates the process.

Fig. 19 is a flowchart showing the process performed by the automatic search expression generation module 26. The automatic search expression generation module 26 initializes the module itself (step S21), and selects one of the dimensions of the input array as a dimension to be processed (step S22).

Next, the automatic search expression generation module 26 generates a search expression by extracting one search condition for each dimension from among search conditions on which input marks are put in the dimension to be processed and from among all the search conditions in the remaining dimensions, and by coupling the extracted search conditions by "AND". Then, the automatic search expression generation module 26 registers the generated search expression to a search array shown in Fig. 20 as a search query

(step S23).

Then, the automatic search expression generation module 26 checks whether or not all of round robin combinations have been extracted when a search condition is extracted from each of the dimensions other than the dimension to be processed (step S24). If all of the combinations have not been extracted yet, the automatic search expression generation module 26 repeats the operations in and after step S23. If all of the combinations have been extracted, the automatic search expression generation module 26 checks whether or not all of the dimensions have been selected as a dimension to be processed (step S25).

If a dimension yet to be selected is left, the automatic search expression generation module 26 repeats the operations in and after step S22. If all of the dimensions have been selected as a dimension to be processed, the automatic search expression generation module 26 provides the search engine 24 with the search queries registered to the search array one by one, requests search execution (step S26), and terminates the process.

. Fig. 21 is a flowchart showing the process performed by the search result switching module 27. The search result switching module 27 first

initializes the module itself (step S31), and receives search results from the search engine 24 (step S32). Next, the search result switching module 24 examines the received search results, obtains the
5 number of hits, related words, and search result lists, which correspond to a search query, and registers them to a search array (step S33).

Then, the search result switching module 27 checks whether or not the search results
10 corresponding to all of search queries are obtained (step S34). If a search result yet to be obtained is left, the search result switching module 27 repeats the operations in and after step S32. If all of the search results are obtained, the search result
15 switching module 27 terminates the process. Then, the information of the search array is passed to the multi-dimensional output module 28 or the visualizing module 29.

Fig. 22 is a flowchart showing the process
20 performed by the multi-dimensional output module 28. The multi-dimensional output module 28 first initializes an output screen (step S41), inputs the display item selected by a user (step S42), and determines which of the number of hits, a related
25 word, and a search result list the display item is

(step S43).

If the display item is the number of hits, the multi-dimensional output module 28 displays the number of hits in each cell in the column or the row, which corresponds to a search condition on which an input mark is put in an input array, in the table on the output screen (step S44). If the display item is a related word, the multi-dimensional output module 28 displays a related word in each cell (step S45). If the display item is a search result list, the multi-dimensional output module 28 displays a result list in each cell (step S46).

Next, the multi-dimensional output module 28 determines whether or not a graph display is instructed (step S47). If the graph display is not instructed, the multi-dimensional output module 28 determines whether or not to continue the output process (step S48). If the multi-dimensional output module 28 determines to continue the output process, it repeats the operations in and after step S42. If the multi-dimensional output module 28 determines not to continue the output process, it terminates the process. Whether or not to continue the output process is determined, for example, based on the instruction from the user.

If the graph display is instructed in step S47, the multi-dimensional output module 28 requests the visualizing module 29 to perform the graph display (step S49), and terminates the process.

5 Fig. 23 is a flowchart showing the process performed by the visualizing module 29. The visualizing module 29 first initializes an output screen (step S51). Then, the visualizing module 29 sets the horizontal (X) and the vertical (Y) axes of
10 a graph to the contents selected by a user (step S52), and inputs the form of the graph selected by the user (step S53). As the form of the graph, a line graph, a bar chart, a pie chart, etc. can be cited. For the pie chart, the contents represented by one
15 circle and those represented by a plurality of circles are selected instead of the contents represented by the horizontal and the vertical axes.

 Next, the visualizing module 29 displays the number of hits corresponding to the selected
20 contents as the selected graph form (step S54), and checks whether or not a portion of the graph has been clicked (step S55). If the graph has been clicked, the visualizing module 29 displays the result list corresponding to the clicked portion (step S56).

25 Then, the visualizing module 29 determines

whether or not the user selects a different graph form (step S57). If the different graph form is selected, the visualizing module 29 repeats the operations in and after step S53. If the different
5 graph form is not selected, the visualizing module 29 determines whether or not to continue the graph display (step S58). If the visualizing module 29 determines to continue the graph display, it repeats the operations in and after step S52. If the
10 visualizing module 29 determines not to continue the graph display, it terminates the process. Whether or not to continue the graph display is determined, for example, based on the instruction from the user.

The search system according to this preferred
15 embodiment can be configured by using an information processing device (computer) shown in Fig. 24. The information processing device shown in this figure comprises a CPU (Central Processing Unit) 121, a memory 122, an input device 123, an output device
20 124, an external storage device 125, a medium driving device 126, and a network connecting device 127, which are interconnected by a bus 128.

The memory 122 includes, for example, a ROM (Read Only Memory), a RAM (Random Access Memory),
25 etc., and stores programs and data used for

processing. The CPU 121 performs necessary processing by executing the programs with the memory 122.

In this case, the search engine 24, the multi-dimensional input module 25, the automatic search
5 expression generation module 26, the search result switching module 27, the multi-dimensional output module 28, and the visualizing module 29, which are shown in Fig. 2B, correspond to the software components implemented by the sets of instructions
10 stored in specific program code segments in the memory 122. Additionally, the input arrays shown in Figs. 17 and 18 and the search array shown in Fig. 20 are arranged in the memory 122.

The input device 123 is, for example, a
15 keyboard, a pointing device, a touch panel, etc., and is used to input the instruction or the information from a user. The output device 124 is, for example, a display, a printer, a speaker, etc., and is used to output a message to a user or a process result.

20 The external storage device 125 is, for example, a magnetic disk device, an optical disk device, a magneto-optical disk device, etc. The information processing device may store the above described programs and data in the external storage device 124,
25 and can use them by loading into the memory 122

depending on need. Additionally, the external storage device 125 may be used also as a text database 23 shown in Fig. 2B.

5 The medium driving device 126 drives a portable storage medium 129, and accesses its stored contents. As the portable storage medium 129, an arbitrary computer-readable storage medium such as a memory card, a floppy disk, a CD-ROM (Compact Disk-Read Only Memory), an optical disk, a magneto-optical disk,
10 etc. is used. A user may store the above described programs and data in the portable storage medium 129, and can use them by loading into the memory 122 as occasion demands.

The network connecting device 127 communicates
15 with an external device via an arbitrary network (line), and performs data conversion accompanying a communication. The information processing device receives the above described programs and data from an external device via the network connecting device
20 127, and can use them by loading into the memory 122 depending on need.

Fig. 25 shows computer-readable storage media which can provide programs and data to the information processing device shown in Fig. 24. The
25 programs and data stored in the portable storage

medium 129 or an external database 130 are loaded into the memory 122. The CPU 121 performs necessary processing by executing the programs with the data.

The above described preferred embodiment refers
5 to the example where documents stored in a text database are searched. The present invention, however, can be applied also to a search performed for arbitrary information such as image data, speech data, a program, etc. By way of example, for an image
10 data search, feature information of a particular color or pattern is considered to be used instead of a keyword. For a speech data search, feature information of a particular voice or sound is considered to be used as a search condition instead
15 of a keyword.

According to the present invention, a user interface which is easy to use and is highly efficient at a search is realized when a plurality of search results for a plurality of search condition
20 combinations are compared in an information search.